SOME EXPERIMENTS AROUND A NEURAL NETWORK FOR MULTIMODAL ASSOCIATIONS

Boniface Yann
Cortex Team
LORIA-Université Nancy 2
Campus Scientifique
BP 239
54506 VANDOEUVRE-lès-NANCY CEDEX
email: Yann.Boniface@loria.fr
http://cortex.loria.fr

Reghis Abdelmalek
Cortex Team
LORIA
Campus Scientifique
BP 239
54506 VANDOEUVRE-lès-NANCY CEDEX
email: reghisabdelmalek@yahoo.com

ABSTRACT

This paper presents a study of the model of triple BAM by [11] which is an improved variation of the original BAM model by [7]. This class of model aims at integrating different sensory inputs in order to memorize a unified and distributed representation. An experimental evaluation of the model is presented that underlines its limitations in terms of noise robustness and learning capacities. A new model is presented in order to overcome those initial limitations by introducing a new online learning algorithm adapted from the PRLAB initial algorithm that improve both noise robustness and learning capacities. Finally, model properties and limitations are considered and discussed within the context of multi-modal integration and brain modeling.

KEY WORDS

Neurons networks, bidirectional associative memory

1 Introduction

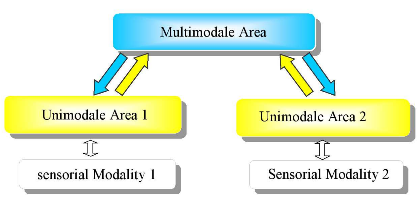

Figure 1. Physiological vue to the multisensorial integration.

The neuromimetism, or connexionnism, is the study of the networks of artificial neurons. The origin of this approach takes as a starting point the cerebral mechanisms in order to develop new paradigms of computation [3].

This work describes an approach of the phenomenon of multimodal integration. Indeed, understanding and control these mechanisms may provide at computer scientists news tools and algorithms to process the heterogeneous inputs signals of their biological inspired models.

To study this phenomenon, we take as a starting point the neuro-physiological definition of the multimodal integration. The results of the electrophysiological studies show that the multimodal integration is built by the convergence of the sensory surfaces in a multimodal common site [1]. These studies proposed a general outline (Figure 1) in which the interaction between the zone of multimodal convergence and the unimodals zones would be at the base of the integration [11].

Our work focus on the algorithmic of this interaction between multimodals areas and unimodals areas and on the learning algorithmic and the topology of the integration zone, which is commonly named associative memory.

2 Associative memories

The memory is a process of storage and exploitation of a knowledge previously acquired, this process takes place on the basis of a modification of the properties of a physical support. In terms of computer sciences simulation, we may use two different techniques:

1. data-processing approach: an access by address, sequential and localist with a static representation
2. connectionist approach: access by contents, parallel and distributed with a dynamic representation.

An associative memory is usually a model which stores a link between a specific output and a specific input, in order to recall the output when the input is presented. An associative memory which associates inputs to themselves is called an auto-associative memory whereas a memory which produces outputs different from the inputs is called hetero-associative memory.

The first work seeking to model the associative memory were primarily interested in its autoassociative properties (like the Hopfield and Kohonen models) [3]. On the other hand, only few models of hetero-associative memories exist.

The work presented in this paper comes from the model of E.Reynaud, primarily inspired by the original model presented by B.Kosko [6][7].

The Kosko Model The BAM's network of Kosko [7] is a model of hetero-associative memory. This model specializes the auto-associative model of Hopfield in order to associate different inputs. The BAM's network is made up of two layers of neurons of different sizes. Each layer simulates a different input of the network. The layers are completely inter-connected, and the weights of the connexions are symmetrical, in a bidirectional way (Figure 2).

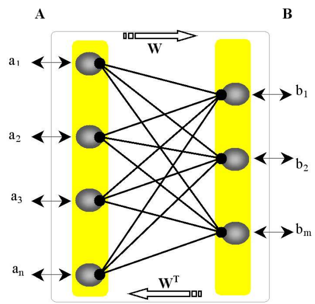

Figure 2. The hetero-associative network (BAM) of Kosko.

When a couple of inputs is presented to the network, the information (the activation of each layer) is reverberated (phase of relaxation) between the two layers of neurons until a state of balance is reached. The weights of the connections of this model are learnt according to the rule of Hebb [3].
One of the principal limits of the BAM is its low storage capacity. This limit is due to the rule of training and the symmetry of connections. In [13], Wasseman shows that the storage capacity of the BAM, when N is the size of the smallest layer, is (N/(2.log2(N))).
Another problem of the BAM, inherited from the hopfield model, is the catastrophic interference.
In order to improve the BAM properties, several methods appeared which seek to increase its storage capacity. These methods explore two main ways: the improvement of the architecture [4][5][12][14][2] and the developement of news learning algorithms [8][15][12][13][16].
Among these algorithmic solutions, E.Reynaud was mainly interested by the method of pseudo-relaxation PRLAB (Pseudo Relaxation Learning Algorithme for BAM) of Oh and Kothari [8].

The Oh and Kothari PRLAB algorithm. The PRLAB algorithm, from Oh and Kothari, is an iterative algorithm which converges with a finished number of steps. This algorithm is based on a variation of the method of relaxation, inspired from a mathematical technique of resolution of system of linear inequations.
According to Oh and Kothari [8], PRLAB provides several advantages, it exploits the maximum capacity of storage of the BAM, it provides a perfect recall for N learned pairs, with N neurons in each layer of the BAM, and a storage of the almost orthogonal examples. Moreover PRLAB is stable and converges quickly.

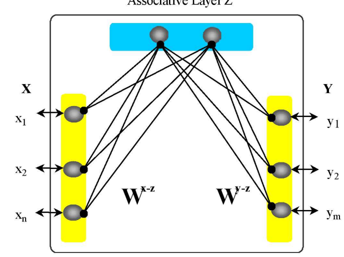

Figure 3. The BAM triple model of E.Reynaud. This model is build with two inputs layers and an associative layer (Z).

The E.Reynaud Model In order to simulate multisensorial integration, E Reynaud -in is Phd Thesis [11]- proposes a new adaptation of the BAM model of Kosko. In her model, E.Reynaud adds an associative layer to the original BAM model, this layer is added between the two different inputs layers (Figure 3). This associative layer (Z) connects the two perceptive layers (X and Y). The connections between the associative layer and the perceptive layers are bidirectional and asymmetric. The learning algorithm is based on PRLAB. The recall of the associated pairs is initiate by the activation of the perceptive layers.
This model, called *triple BAM*, is an connectionist alternative solution to store associations of inputs patterns. In addition to the storage, E.Reynaud wishes to integrate the inputs in the associative layer, the pattern matching learnt in the associative layer can be regarded as a *linking code* of the perceptions.

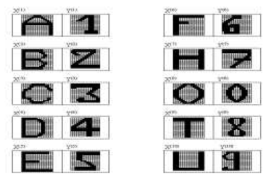

Figure 4. The ten matching set of the test data base.

To evaluate the properties and the capacities, in terms of learning and generalization, of this new model of BAM, we have used the data base of E Reynaud. This data base consists of images of letters and images of numbers (Figure 4): the letters are encoded by the perceptive layer X out of 256 neurons, and the numbers by the perceptive layer Y on 225 neurons.
To initiate the learning, the pattern presented at the associative layer (Z in the figure) is a random pattern, which has to be a discriminant pattern. In our experimental network, this pattern is encoded on 150 neurons.

Conclusions of our experimentations After our experiments, we isolated some problems which penalize the performances and the robustness of the triple BAM model:

- The initialization of the associative layer acts upon the noise robustness of the triple BAM.
- The order of presentation of the matching patterns acts on the learning.

Moreover, this model presents two main limits:

- The Oh and Kothari algorithm is a batch learning algorithm, it is impossible to make a dynamical learning of a new pattern matching (on-line learning).
- The triple BAM has low performances in terms of recalls with the lack of one modality (perception).

The goal of this work is to find solutions to solve these different limits in order to approach the phenomenon of multimodal integration. We particularly wish to focus on the distributed character of computations, on the results and the robustness of this hetero-associative memory.
The next sections of this paper presents three new algorithms to improve the Bam of E.Reynaud and which preserve the main properties of her model: the architecture and the learning algorithm of the triple BAM.

3 The multiassociative BAM

The linking code initialization The learning algorithm of the triple BAM needs a discriminant initialization of the associative layer to converge, a discriminant *linking code*. Moreover, our studies show a strong influence of this initialization on the dynamic of the network:

- Random and discriminant initialization make the network noise sensitive.
- An initialization based on the compression of the inputs makes best performances for the dynamic. But this method doesn't assure the discrimination and so this doesn't assure the convergence of the learning.

We provide here a solution to assure the convergence of the learning and the noise robustness of the network. This solution uses *affectation areas* to initialize the associative layer (Figure 5). With this initialization method, for a N

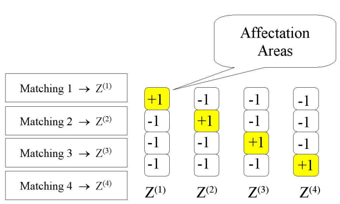

Figure 5. Associative layer initialization with the affectation areas method.

neurons associative layer, and an one neuron length *affectation area*, we may affect N different linking codes. And so, it is possible to learn N different matching.
This method assures a hamming distance higher than two times the length of the affectation areas. And each *linking code* is different, so it assures the convergence of the learning.
On the other hand, we lost the distributed aspect of the coding of the associative layer.

The architecture of the multiassociative BAM The *multiassociative BAM* is an adaptation of the *triple BAM*. The main evolutions are:

- The associative layer is initialized with the affectation areas method.
- The order of presentation of the matching patterns is random.

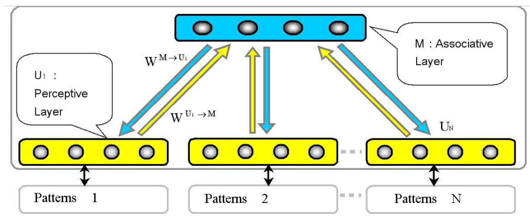

Figure 6. The architecture of the multiassociative BAM

In terms of architecture, the multiassociative BAM is composed with two different kinds of layers:

- One input layer per perception $(U_1, ..., U_n)$.
- The multimodal associative layer (M).

The learning algorithm is issue to the PRLAB algorithm of the *triple BAM* model.

Evaluation of the affectation areas method. In this section we evaluate the influence of the initialization of the associative layer, and more particularly, the results of the *affectation areas* method. To do this, we made the experiments on a model with two perceptives layers : the first (X) contains 256 neurons and the second (Y) 225 neurons. The multimodal associative layer (Z) contains 150 neurons.

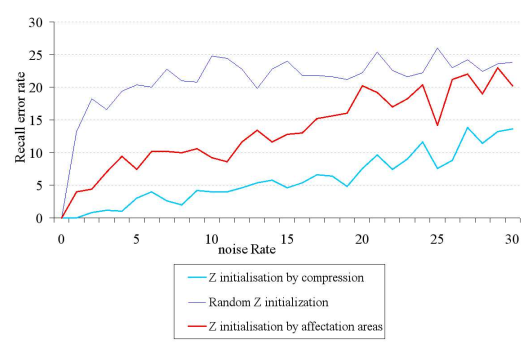

Figure 7. The error rate for the recall of pattern matching.

The results (figure 7) show that the new method for initialization, by the use of affectation areas, improves the noise robustness of our network. Indeed, if the method of compression is better than affectation areas, in terms of the noise robustness, the compression does not assure the convergence of the learning. On the other hand, the method of random initialization, which assures convergence, is worse than affectation areas in terms of the noise robustness.

The influence of the random pulling of the inputs. To avoid results dependent on a specific data base, we use a 50 patterns matching data base. Each pattern of this base is coded on 50 neurons, built with $+1$ and -1 randomly distributed on the neurons. Our *multiassociative Bam* is built with two perceptive layers, with 50 neurons each, and an 50 neurons associative layer.

We want to study the learning and recall abilities of our *multiassociative BAM* when the examples used are randomly selected. To do that, our model has learnt an increasing number of random patterns (from 10 to 50 patterns). And, for each step, we have calculated the error rate of the recall on the perceptive layers. And, for each step, we have calculated the error rate of the recall on the perceptive layers, after convergence.

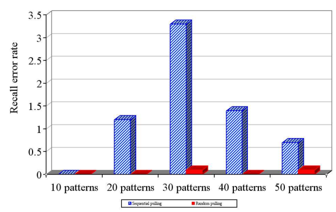

Figure 8. Error rate for the recall

Figure 8 shows the results after 100 experiments. The random pulling of the inputs strongly improve the learning performances of the *triple BAM* (the error rates are lower than 0.5%).

The noise robustness The recall capacities, from noisy data, are important properties from the associative memories. We measure here the error rates, in terms of recall in the perceptive layers, when one input is noised (X) and then when both (X and Y) are noised.

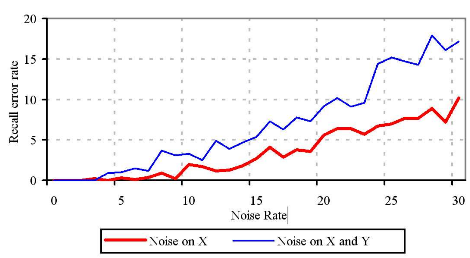

Figure 9. Recall error rate with noisy data.

Figure 9 shows the results after 100 experiments. We may conclude that our model is robust in terms of noisy data. The recall error rate stays lower than 10% even for 30% noised pattern.

Recall with missing data. We evaluate here the capacity of recall with missing data of the *multiassociative BAM*. We have tested the recall of a matching pattern when one input is missing. When a white pattern replace the missing input, as shown in figure 10, the recall error rate is higher than 11%.

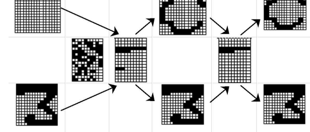

Figure 10. Example of a recall with a missing input, replaced by a white pattern.

Nevertheless, the choice of the algorithmic representation of the missing pattern is problematic. This representation leads to present one pattern, same if it is totally white, to the perceptive layer, this is not really one simulation of a missing pattern.

Conclusions for *multiassociative BAM*. To develop the *multiassociative BAM* we have made some modifications on the *triple BAM*. We have modified the initialisation of the associative layer, the adaptation of this associative layer and the draw of the examples during the learning phase.

These modifications really improve the noisy robustness and the recall capacities of the original *triple BAM* model.

Nevertheless, the learning algorithm of the *multiassiciative BAM* is the PRLAB algorithm of Oh and Kothari [8]. This algorithm, which improves the storage capacities of the BAM models, is off-line by design. The PRLAB algorithm, which is used in our model, needs to know all the example corpus to begin the learning. It is currently impossible to learn new perceptions dynamically, and to have a network which adapts its weights during its interactions with its environment.

4 The *on-line multiassociative BAM*.

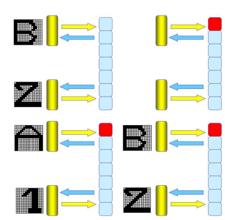

Figure 11. The *on-line multiassociative BAM* learning algorithm. When a new pattern matching, (B-2) here,is presented to the network and is not recognized (network's relaxation doesn't converges on an affectation area). The network activates each its stored affection areas, here only one affectation area, this activation recalls the learning pattern matching, here (A-1). (B-2) is not learned, it is not close to a learning pattern matching then the network creates a new affectation area for this pattern matching.

To approach biological reality, and with the Cortex team's perspectives in terms of autonomous robotics, we are interested to develop models which may learn dynamically (*on-line* algorithms). To implement this kind of learning, we may decide on a compromise between adaptation

and stability. The network may forgot some already learned data, with too much adaptation or it may be unable to learn the new data, with too much stability.
As we saw, the PRLAB learning algorithm doesn't allow the on-line learning. To provide on-line learning property to our network, we need to adapt PRLAB to define a new learning algorithm.
We propose a new method based on the stable states of the network. The stable states are the states of the network after it has converged for a pattern matching. For each new pattern, or pattern matching, the network tests all its stable states and decide if the new input must be learnt.
The study of the stable states is based on the *inverse recall*. After the learning of a pattern, we store the state of the associative memory which is associate to this pattern. With the affectation areas, this step consists of the storage of the affectation area concerned by this input pattern (Figure 5).

When a new pattern matching is presented to the network, the *on-line multiassociative BAM* activates the associative layer with each stable state stored. After relaxation, the perceptive layers are activated. Our algorithm calculates distance between these activities, which represents patterns already learnt, and the new inputs. If this distance is sufficient, the network learns this new pattern in a new affectation area (Figure 11).
Learning a new pattern consists on the store of the new pattern (the sum of the perceptive inputs) with the patterns already learned, then we use PRLAB algorithm with the complete corpus of the patterns stored.
This method allows to not forget the patterns previously learned, because they are all included in the news PRLAB learning process.

Experiments with the *on-line multiassociative BAM*. We have studied the properties of this new learning algorithm for the multiassociative BAM. This study concerns the learning capacities of the algorithm and the recall capacities of the network.
We tested our model with a 50 patterns data base. We used a network with two perceptive layers, with 50 neurons each. The associative layer contains 50 neurons too. The *on-line multiassociative BAM* learnt an increasing number of random patterns. For the tests, the model learnt an increasing number of random patterns. The affectation areas length of the *on-line multiassociative BAM* is set to 5 neurons for the 10 patterns matching tests, and it to 2 neurons for the 20 patterns matching tests. The results obtained after 100 experiments are shown in table 1.

	10 patterns learnt		20 patterns learnt	
	mean	standart deviation	mean	standart deviation
epochs	3.00	0.00	5.40	1.01
'reverberations'	1.11	0.17	1.31	0.22
Recall error rate	**0.30**%	0.64%	**1.30**%	1.45%

Table 1. learning results with the on-line multiassociative BAM (PRLAB parameters used: $\lambda = 1.9 \, and \, \xi = 50$).

These results show that the model may learn 20 patterns matching with a recall error rate of $1.30\,\%$. These experiments have also shown the limits of our model. Some experiments with more than 25 patterns matching as inputs, with the affectation areas set to 1 neuron, the learning fails. Our results show that the storage capacity of the model is lower than $0.5 * N$ different inputs, when the associative layer contains N neurons.

Conclusions for the *on-line multiassociative BAM*. The *multiassociociative BAM*, with the on-line adaptation of the PRLAB algorithm, may adapt his configuration (its weights) to new perceptive signals (inputs), without forgetting its previous learning. Thus, this solution solves one of the main weaknesses of the triple BAM model, in terms of biological plausibility. The initialisation of the associative layer with affectation areas and the recall of the stable states are the main tools to obtain this property.
Nevertheless, we need to know some parameters to use our algorithm: the maximum number of examples to learn and the length of the affectation areas of the associative layer. Moreover, the value of these two parameters are a great influence on the learning performances of our network. The on-line algorithm doesn't solve by itself the problem of the compromise between adaptation and stability. That requires an external tuning of the parametres.
On the orther hand, the algorithm doesn't use the length of the associative layer. Each new example learnt just adds the length of the affectation area in the number of neurons used of this layer. With this property, it seems possible to convert our algorithm to an incremental learning algorithm. Thus, the associative layers will only contain the used neurons and each new learning will increase the length of the associative layer.

5 Conclusions

We have presented in this paper several experiments related to the triple BAM model as introduced by E. Reynaud in order to evaluate the model in terms of information processing based on distributed and numerical computations. In the context of adaptive algorithms, it is quite clear that this model lacks a unified representation of the environment would enable for example a robot to evoluate freely in an unknow environment.

This study clearly demonstrates that performances of the model concerning noise robustness and learning are not suitable for online learning, mainly because there is a strong dependency between learning and the order of presentation of examples as well as a weakness in recall when a whole perception is missing. This has been underlined throughout the study and emphasize the need for alternative algorithms.

To cope with these limitations, we introduced some original algorithms in order to improve the triple BAM model. More precisely, the new initialization algorithm that has been introduced for the associative layer and the initialization by affectations areas greatly enhanced noise robustness while ensuring the convergence of learning. Moreover, the new random distribution of examples provides a simple solution to the dependency problem and the online learning algorithm, based on the one by [8] allows for an extended use of the affectation areas method. Finally, while this was not the original goal of the study, it is to be noted that global performances of the original model have been greatly improved.

If we now look closer to the structure of the model, it is clear that the model we introduced may be seen as a model of the hippocampus instead of a multi-sensory integration cortical area. If we look at experimental results, it is quite clear that the model exhibits declarative memory properties (i.e. learning by heart) where each new example can either be learned or recalled. Consequently, This functional behavior is not realloy suited for generalization

Furthemore, affectation areas that have been used to initialize the associative layer, have been introduced in the first place to solve the convergence problem of the PRLAB learning algorithm. The problem with this approach is that the associative layer does not represent any more a real integration of presented data. This layer is now more of a kind of pointer to a learned example where the on-line algorithm is used to somehow seek something within memory. This is problematic when one wants to have a real distributed model: information is not distributed anymore but is localised within affectation areas.

Finally, we want to emphasize that the triple BAM model is not adapted to real distributed and numerical computations. The problem is not architectural but rather functionnal in terms of offline learning where the PRLAB algorithm revealed itself not adapted at all. Being rooted in the Hopfield learning algorithms (which is totally offline), it is not designed to provide a continuous learning. Furthermore, PRLAB has been designed to provide performances in terms of storage capacities and not to provide a unified representation of perceptions, that is still to be done.

References

[1] Calvert, G., Hansen, P., Iversen, S., and Brammer, M. Detection of audio-visuel intgration sites in humans by application of electrophysiological criteria to the BOLD effect. Neuroimage, 14(2) :427-7438, 2001.

[2] M. Hagiwara, Multidirectional Associative Memory, Porc. IEEE Joint Conf. On Neural Networks, Washington, D.C, January 15-19, 1990. New York : IEEE, Vol I, pp. 6-6, 1990.

[3] J. Hertz, A. Krogh et R. G. Palmer. An Introduction to the Theory of Neural Computation. Lecture notes Volume I, Addison Wesley, 1991.

[4] B. Humpert. Bidirectional associative memory for three patterns. In International neural network conference, Kluwer academic publishers. volume 2, pages 843-851, Paris, July 1990.

[5] H. Kang, Multilayer associative neural network (MANNs): Storage capacity versus perfect recall, IEEE Transactions on Neural Networks, vol. 5, no. 5, pp. 812-822, September 1994.

[6] B. Kosko, Adaptive bidirectional associative memories, Applied Optics, vol. 26, no. 23, pp. 4947-4960, 1987.

[7] B. Kosko, Bidirectional associative memories, IEEE Transactions on Systems, Man, and Cybernetics, vol. 18, no. 1, pp. 49-60, 1988.

[8] H. Oh and S. C. Kothari, Adaptation of the relaxation method for learning in bidirectional associative memory, IEEE Transactions on Neural Networks, vol. 5, no. 4, pp. 573-583, July 1994.

[9] A. Reghis Un réseau de neurones pour des associations multimodales. M.Sc.Thesis, Université Nancy 1 - Henri Poincaré, Nancy, France, 2003.

[10] A. Reghis, F. Alexandre, and Y. Boniface, Un réseau de neurones pour des associations multimodales, 14ème Congrès Francophone AFRIF-AFIA de Reconnaissance des Formes et Intelligence Artificielle - RFIA'2004, Toulouse, France, 2004.

[11] E. Reynaud, Modélisation connexionniste d'une mémoire associative multimodale. PhD. Thesis, Institut National Polytechnique de grenoble, 2002.

[12] C.-C. Wang and H.-S. Don, An analysis of high-capacity discrete exponential BAM, JEEE Transactions on Neural Net- works, vol. 6, no. 2, pp. 492-496, 1995.

[13] Wasserman, P. D., Neural computing : Theory and practice. New York. 1987.

[14] Z.-B. Xu, Y. Leung, and X.-W. He, Asymmetric bidirectional associative memories, IEBE Transactions on Systems, Man, and Cybernetics, vol. 24, no. 10, pp. 1558-1564, 1994.

[15] Z. Wang, A bidirectional associative memory based on optimal linear associative memory, IEEE Transactions on Computers, vol. 45, no. 10, pp. 1171-1179, 1996.

[16] X. Zhuang, Y. Huang, and S.-S. Chen, Beter learning for bidirectional associative memory, Neural Networks, vol. 6, no. 8, pp. 1131-1146, 1993.